

```
*****
* This program will calculate the recodes for the public use file. AHS National 1997
*
* Program: pufrcd.sas
* Programmed by: Diane Schwartz
* Date: January 20,1998
* Updated: April 9,1998
* Revised: May 19,1998 - corrected universe problem for zsmhc
* Revised: November 20, 1998 - included a new recode for HISTRY
* Revised: January 27, 1999 - corrected nummor according to new specs
* Revised: April 8, 1999 - corrected problem with nummor, not counting all
*           the cases that have no mortgage.
* Revised: April 20, 1999 - added new variable METRO2, because of a problem
*           with no geography conversion for 90 - 80 on
*           UACODE. This new variable is a recode of METRO.
* Revised: May 24, 1999 - added new variable METRO3 and took out the
*           variable METRO2
*
* Input Library: /natdata/(survey year)/topcodes
*                 /natdata/(survey year)/bridge
*                 /natdata/(survey year)/master.ssd01
*
* Output Library: /natadata/(survey year)/repcodes
*
* The following recodes are coded in this program:
*   cmsa - renamed from cmsa80
*   smsa - renamed from pmsa80
*   per
*   zadult
*   zinc
*   zinc2
*   kitchen
*   plumb
*   rooms
*   confee
*   zadeq
*   zmshc
*   amtx
*   nummor
*   ran
*   rac
*   degree
*   metro
*   histry
*   metro2 -- as of May 24, 1999 this recode is no longer done
*   metro3
```

```

/*
***** set up lib names ** */
libname bridg "$BRIDGE";
libname rcd  "$RECD";
libname mast "$MASTR";
libname topc "$TOPCD";

missing D R B X M N U _;

/* sort datasets on control number 13 to enable views to be created */
/*
proc sort data=topc.houshld out=work.houshld;
  by ctrlnm13;
run;

proc sort data=topc.person out=work.person;
  by ctrlnm13 person;
run;

proc sort data=mast.master out=work.master;
  by ctrlnm13;
run;

proc sort data=topc.homimp out=work.homimp;
  by ctrlnm13;
run;

proc sort data=topc.mortg out=work.mortg;
  by ctrlnm13;
run;

proc sort data=bridg.homimpr out=work.homimpr;
  by ctrlnm13;
run;
*/
/* ** create a view to process the recodes ** */
data viewed1 / view=viewed1;
  merge topc.houshld (in = hhldend
    keep = ctrlnm13 amte amtf amtg ami amto amtt amtw
      amtxq amtxre badstep bedrms bigp burner busin
        camfq climb confeeq cook cracks dens dining elev
          famrm flrent fmhotf freeze frent hequip prent
            holes hotpip ileak kexclu kitch leak living
              lrrent ltsok m3rod mhotfe mice nowire numblow
                
```

```

        numcold numltl nunit2 othfn oven plugs
        railok rats recrm refr rent sharpf sink
        status tenure toilet tub txre vacancy
        vother curper buye cstmnt baths)
topc.person (in = endper
            keep = ctrlnm13 age hhmem rel sal);
by ctrlnm13;
if not (endper and hhldend) then
do;
  if endper then
  do;
/* put 'no houshld obs for persons ctrlnm13 = ' ctrlnm13; */
  cnohh+1;
end;
else
do;
/* put 'no persons obs for houshld ctrlnm13 = ' ctrlnm13; */
  cnopr+1;
end;
end;
run;

```

```

data viewed2 / view=viewed2;
merge mast.master (in=endmst
                  keep = ctrlnm13 cmsa80 region pmsa80 coolday
                           heatday msasta80 uacode80 urbrur80
                           convdate
                           rename = (cmsa80=cmsa pmsa80=smsa))
                  viewed1 (in=endvw);
by ctrlnm13;
if not (endmst and endvw) then
do;
  if endvw then put 'no master obs for viewed ctrlnm13 = ' ctrlnm13;
  else delete;
end;
run;

```

```

data viewed3 / view = viewed3;
merge topc.homimp (in=hiend
                  keep = ctrlnm13 adbedam adbtham addbed addbth addfixt
                           addfuse addinsl addkit addoth addpipe addrwn
                           addsid addwire adfinfl adfloor adflram adftam
                           adheam adinsam adkitam adooram adotham adpipam
                           adsidam adsibam adsubfl adwiram allpipe allsid

```

```

alpipam alsidam bthcabn bthcbam bthcntr bthctam
bthelam bthelec bthflam bthlite bthltam
bthpnt bthptam bthsink bthskam bthtlam bthtoil
bthtub bthwall bthtbam bthwlam bucprt budeck
bugarag buothot buporch crptam deckam dispam
driveam drivewy dshwam dsplam fence fenceam
finflam fuseam garagam hequam hteqins
insaam insbam inscam kitapam kitappl kitcavn
kitcbam kitcntr kitctam kitelam kitelec kitflam
kitflr kitlite kitltam kitpnt kitptam kitsink
kitskam kitwall kitwlam newpanl nwbed nwbedam
nwbth nwbtham nwkit nwkitam nwoth nwotham
nwpnlam nwrec nwrecam othotam outaam outbam outcam
patio patioam pool poolam porcham rcenam renbedr
renoth repfixt repfuse rephequ repinsl replcen
reppanl repsept repwtrh rewiram rewire rnbedam
rnotham roof roofam rpanlam rpdoram rpdrwn
rpdshts rpdspsl rpfxtam rpinsam sdflr sdflram
septam shed shedam wtrham quake tornado lndsl
fire flood othdis)
viewed2 (in=endvw2);
by ctrlnm13;
if not (hiend and endvw2)then
do;
  if hiend then put 'no viewed2 obs for homimp ctrlnm13 = ' ctrlnm13;
end;
run;

data viewed4 / view = viewed4;
merge bridg.homimpr (in=hirend keep = misina misinb
                     misinc misouta misouth misoutc ctrlnm13)
      viewed3 (in=endvw3);
by ctrlnm13;
if not (hirend and endvw3) then
do;
  if hirend then put 'no viewed3 obs for homimpr ctrlnm13 = ' ctrlnm13;
end;
run;

data viewedt / view = viewedt;
merge topc.mortg (in=mtgend
                  keep = ctrlnm13 amtm amtm2 henum hepmt1 hepmt2 hepmt3
                         inspmt inpmt2 pmt pmt2 pmt3 pmt4 ramapp regmor
                         taxpmt txpmt2 mcnt hel henum)
      viewed4 (in=endvwd);

```

```

by ctrlnm13;
if not (mtgend and endvwd) then
do;
  if mtgend then put 'no viewedt obs for mortg ctrlnm13 = ' ctrlnm13;
end;
run;

```

```

/* ** this section creates a data set with the recodes included. ** */
data rcd.pufrcd (keep = ctrlnm13 per zadult zinc zinc2 kitchen plumb
                   rooms confee zadeq zsmhc amtx nummor ran rac
                   region smsa cmsa degree metro histry metro3);

```

```

length ctrlnm13 $13 per 3. zadult 3. zinc 5. zinc2 5. kitchen $1
      plumb $1 rooms 3. confee 3. zadeq $1 zsmhc 4. amtx 4. nummor 4.
      ran 4. rac 5. region $1 smsa 4. cmsa $2 degree $1 metro $1
      histry $1 srvyr $4 metro3 $1;

```

```

set viewedt end=send;
  by ctrlnm13;

```

```

retain incnt 0;
retain ctrlno 0;
retain ctllst 0;
retain outcnt 0;

```

```

incnt+1;
/*  if(mod(incnt,500) = 0) then put 'processing obs number ' incnt; */
  if first.ctrlnm13 then ctrlno+1;
  if last.ctrlnm13 then ctllst+1;

```

```

if first.ctrlnm13 then

```

```

do;
  per = .B;
  zadult = .B;
  zinc = .B;
  zinc2 = .B;
  rooms = .B;
  confee = .B;
  zsmhc = .B;
  amtx = .B;
  nummor = .B;
  ran = .B;
  rac = .B;
  plumb = 'B';

```

```

zadeq = 'B';
kitchen = 'B';
metro = '';
metro3 = '';
end;

if status = '1' then
do;

    if first.ctrlnm13 then
        do;
            per = 0;
            zadult = 0;
            zinc = 0;
            zinc2 = 0;
            i = 1;
            if (-999998 <= vother <= 999998) then
                do;
                    zinc+vother;
                    zinc2+vother;
                end;
            end;
        end;
    end;

/* per and zadult recodes - number of persons and number of adults over 18 */

    if hhmem='1' then
        do;

            per+1;
            if (age >= 18) then
                do;
                    zadult+1;
/*           if ctllst < 10 then
                put 'zadult = ' zadult ' and age = ' age; */
            end;
        end;
    end;

/* zinc recode - income of reference person and household
members related to reference person */

    if (age >= 14 and (rel = 1 or rel = 2 or rel = 20 or (rel >= 22 and
rel <= 26))) then
        zinc+sal;

/* zinc2 recode - household income */

```

```

if (age >= 14) then
    zinc2+sal;

end; /* end of hmem check */

end; /* end of status check */

if status > '0' and status < '4' then
do;

/* kitchen recode - complete kitchen facilities */
kitchen = ' ';

if(((nunit2 = '1' or nunit2 = '2' or nunit2 = '4')
    and (sink = '1')
    and (refr = '1')
    and (cook = '1' or burner = '1' or oven = '1'))
or
((nunit2 = '3' or nunit2 = '5')
    and (kexclu = '1')
    and (sink = '1')
    and (refr = '1')
    and (cook = '1' or burner = '1' or oven = '1'))) then
do;
    kitchen = '1';
end;
else
do;
    if ((sink = '2')
        or (refr = '2')
        or (cook = '2' and burner = '2' and oven = '2')
        or ((nunit2 = '3' or nunit2 = '5') and (kexclu = '2'))) then
        do;
            kitchen = '2';
        end;
    end;
end;

/* plumb recode - exclusive use of plumbing */
plumb = ' ';

if ( hotpip = '1' and toilet = '1' and tub = '1' and sharpf = '2') then
do;
    plumb = '1';
end;
if ( hotpip = '2' or toilet = '2' or tub = '2' or sharpf = '1')then

```

```

do;
  plumb = '2';
end;

/* rooms recode - number of rooms */

rooms = 0;
rooms = sum(bedrms,kitch,living,dining,famrm,recrm,dens,busin,othfn);

if rooms >= 21 then rooms = 21;

/* confee recode - condominium or homeowners assoc. or mobile home monthly
   fees */

if confeeq = . then confee = .;
if confeeq = 0 then confee = 0;
if confeeq >= 1 then
  do;
    confee = (confeeq * camfq) / 12;
    confee = confee - 5;
    confee = round(confee,10);
    confee+5;
  end;
end; /* end of status check */

/* zadeq recode - adequacy of housing */
if status = '1' then
  do;
    zadeq = ' ';
    if (baths < 2 and (hotpip = '2'
      or tub = '2'
      or toilet = '2'
      or sharpf = '1'))
    or
      (freeze = '1'
      and (numcold >= '3' and numcold <= '8'))
    or
      (buye = '1')
    or

```

```
(nowire = '2'  
and plugs = '2'  
and (numblow >= '3' and numblow <= '8')) then  
do;  
    zadeq = '3';  
end;
```

```
n = 0;  
if leak = '1' then n+1;  
if ileak = '1' then n+1;  
if holes = '1' then n+1;  
if cracks = '1' then n+1;  
if bigp = '1' then n+1;  
if rats = 'X' then n+1;  
if n >= 5 then  
    zadeq = '3';
```

```
m = 0;  
if ltsok = '4' or ltsok = '5' then m+1;  
if badstep = '2' then m+1;  
if railok = '1' or railok = '3' then m+1;  
if climb >= 3 and elev ne '2' then m+1;  
if m = 4 then  
    zadeq = '3';
```

```
if zadeq ne '3' then  
do;  
    if numltl >= '3' and numltl <= '8' then  
        zadeq = '2';  
    if hequip = 7 then  
        zadeq = '2';
```

```
n=0;  
if leak = '1' then n+1;  
if ileak = '1' then n+1;  
if holes = '1' then n+1;  
if cracks = '1' then n+1;  
if bigp = '1' then n+1;  
if rats = 'X' then n+1;  
if n = 3 or n = 4 then  
    zadeq = '2';
```

```
m=0;  
if ltsok = '4' or ltsok = '5' then m+1;  
if badstep = '2' then m+1;  
if railok = '1' or railok = '3' then m+1;
```

```

if climb >= 3 and elev ne '2' then m+1;
if m = 3 then
  zadeq = '2';

if kitchen = '2' then
  zadeq = '2';
end;

if zadeq ne '3' and zadeq ne '2' then
  zadeq = '1';
end; /* end of status check for zadeq universe */

```

/* zsmhc - monthly housing costs recode */

```

if amtxre < 0 then amtxre = 0;
if mhotfe < 0 then mhotfe = 0;
if cstmnt < 0 then cstmnt = 0;

```

```

if status = '1' then
do;

```

```

zsmhc = 0;

```

```

if (1 <= amte <= 998) then zsmhc+amte;
if (1 <= amtg <= 998) then zsmhc+amtg;
if (amto >= 4) then zsmhc+(amto / 12);
if (amtf >= 4) then zsmhc+(amtf / 12);
if (amtt >= 4) then zsmhc+(amtt / 12);
if (amtw >= 4) then zsmhc+(amtw / 12);

```

```

if ((amtxq >= 0) and (taxpmt ne '1' and txpmt2 ne '1')) then
do;
  zsmhc+((amtxq - amtxre) / 12);
end;

```

```

if (amti >= 1) and (inspmt ne '1' and inpmt2 ne '1') then
do;
  zsmhc+(amti / 12);
end;

```

```

if (confeeq >= 1) then
  zsmhc+((confeeq*camfq)/12);
if ( 1 <= lrent <= 1996) then
  zsmhc+((lrent * flrent) / 12);
if (nunit2 = '4' or nunit2 = '5') then

```

```

do;
if (mhotfe >= 1) then
  zsmhc+((mhotfe * fmhotf) / 12);
end;
if ((rent >= 2) and (rent < 1 or rent > 9997)) then
  zsmhc+((rent*frent)/12);
if (rent >= 1 and frent >= 1) then
  do;
    zsmhc+((rent*frent) / 12);
  end;
if (pmt >= 1) then
  zsmhc+pmt;
if (pmt2 >= 1) then
  zsmhc+pmt2;
if (pmt3 >= 1) then
  zsmhc+pmt3;
if (pmt4 >= 1) then
  zsmhc+pmt4;
/* if (hepmt1 >= 1) then
   zsmhc+hepmt1; */
/* if (hepmt2 >= 1) then
   zsmhc+hepmt2; */
/* if (hepmt3 >= 1) then
   zsmhc+hepmt3; */
if (amtm >= 1) then
  zsmhc = zsmhc-(amtm / 12);
if (amtm2 >= 1) then
  zsmhc = zsmhc-(amtm2 / 12);
if tenure = '1' then
  do;
    if ((cstmnt / 12) >= 1) then
      zsmhc+(cstmnt / 12);
  end;

zsmhc = round(zsmhc,1);
end; /* end of universe check for zsmhc */

/* amtx recode - yearly real estate taxes */

amtix = .;
if((status = '1' and tenure = '1') or ((status = '2' or status = '3') and
(vacancy = 3 or vacancy = 5))) then
  do;
    if ((status = '1' and tenure = '1') and txre = '2' and amtxq = 0) then
      amtx = 0;
    else

```

```

if((status = '1' and tenure = '1') and txre = '1' and
((amtqx - amtxre) <= 0)) then
    amtx = 0;
else
    if((status = '2' or status = '3') and (vacancy = 3 or vacancy = 5)
        and amtxq = 0) then
            amtx = 0;
    else
        if((status = '1' and tenure = '1') and txre = '2' and amtxq >= 1)
            then
                amtx = amtxq;
        else
            if((status = '1' and tenure = '1') and txre = '1' and
                ((amtqx - amtxre) >= 1)) then
                    amtx = amtxq - amtxre;
            else
                if((status = '2' or status = '3') and (vacancy = 3 or
                    vacancy = 5) and amtxq >= 1) then
                        amtx = amtxq;

        if amtx >= 1 then
            do;
                amtx = amtx - 50;
                amtx = round(amt,100);
                amtx+50;
            end;
    end;
end; /* end of universe check for amtx */

```

/* nummor recode - number of mortgages */

```

if (status = '1' and tenure = '1') then
do;
    nummor = .;
    hcmt = .;

    if ((ramapp = '2' or ramapp = 'D' or ramapp = 'R') and
        mcnt = 0 and hel = '2')then
        nummor = 0;
    else
        do;
            if (ramapp = '1') then
                nummor = 1;
            else
                do;

```

```

if hel = '2' then
  hcnt = 0;
if hel = '1' and (1 <= henum <= 100) then
  hcnt = henum;
if (0 <= mcnt <= 7) and (0 <= hcnt <= 100) then
do;
  nummor = mcnt + hcnt;
  if nummor >= 7 then
    nummor = 7;
  end;
end;
end;
if nummor < 0 and ((hel = 'D' or hel = 'R' or hel = ' ') or
(henum = .D or henum = .R or henum = .)) then
  nummor = .D;

end; /* end of universe check for nummor */

/* ran recode - repair and alteration number */

if status = '1' and tenure = '1' then
do;
  ran = 0;
  if (quake = 'X' or tornado = 'X' or lndslid = 'X' or
  fire = 'X' or flood = 'X' or othdis = 'X') then
    ran+1;

  if nwbtth = 'X' then ran+1;
  if nwbed = 'X' then ran+1;
  if nwkit = 'X' then ran+1;
  if nwrec = 'X' then ran+1;
  if nwoth = 'X' then ran+1;

  if addbth = 'X' then ran+1;
  if addkit = 'X' then ran+1;
  if addbed = 'X' then ran+1;
  if addoth = 'X' then ran+1;

  if bugarag >= '1' and bugarag <= '3' then ran+1;
  if buporch >= '1' and buporch <= '3' then ran+1;
  if budeck >= '1' and budeck <= '3' then ran+1;
  if bucprt >= '1' and bucprt <= '3' then ran+1;
  if buothot >= '1' and buothot <= '3' then ran+1;

  if bthwall = 'X' then ran+1;
  if bthcavn = 'X' then ran+1;

```

```

if bthflr = 'X' then ran+1;
if bthcntr = 'X' then ran+1;
if bthtoil = 'X' then ran+1;
if bthtub = 'X' then ran+1;
if bthsink = 'X' then ran+1;
if bthlite = 'X' then ran+1;
if bthelec = 'X' then ran+1;
if bthpnt = 'X' then ran+1;

if kitwall = 'X' then ran+1;
if kitcabn = 'X' then ran+1;
if kitflr = 'X' then ran+1;
if kitcntr = 'X' then ran+1;
if kitappl = 'X' then ran+1;
if kitsink = 'X' then ran+1;
if kitlite = 'X' then ran+1;
if kitelec = 'X' then ran+1;
if kitpnt = 'X' then ran+1;

if renbedr = 'X' then ran+1;
if renoth = 'X' then ran+1;
if roof = '1' then ran+1;

if addsid = 'X' then ran+1;
if allsid = '1' or allsid = '2' then ran+1;
if addpipe = 'X' then ran+1;
if allpipe = '1' or allpipe = '2' then ran+1;
if addwire = 'X' then ran+1;
if rewire = '1' or rewire = '2' then ran+1;
if addfuse = 'X' then ran+1;
if repfuse = 'X' then ran+1;
if addrwn = 'X' then ran+1;
if rpdrwn = 'X' then ran+1;
if addfixt = 'X' then ran+1;
if repfixt = 'X' then ran+1;
if addinsl = 'X' then ran+1;
if repinsl = 'X' then ran+1;
if adsubfl = 'X' then ran+1;
if adfinfl = 'X' then ran+1;
if adfloor = 'X' then ran+1;
if sdflr >= '1' and sdflr <= '3' then ran+1;

if newpanl = 'X' then ran+1;
if reppanl = 'X' then ran+1;
if replcen >= 1 and replcen <= 3 or replcen = .D or replcen = .R then
    ran+1;

```

```

if rephequ = 1 then ran+1;
if hteqins = 1 or hteqins = 2 or hteqins = .D or hteqins = .R then
    ran+1;
if repsept = 1 or repsept = 2 then ran+1;
if repwtrh = 1 or repwtrh = 2 then ran+1;
if rpdshws = 1 or rpdshws = 2 then ran+1;
if rpdspsl = 1 or rpdspsl = 2 then ran+1;

if misina ne '' then ran+1;
if misinb ne '' then ran+1;
if misinc ne '' then ran+1;

if drivewy = '1' then ran+1;
if fence = '1' then ran+1;
if patio = '1' then ran+1;
if pool = '1' then ran+1;
if shed = '1' then ran+1;

if misouta ne '' then ran+1;
if misoutb ne '' then ran+1;
if misoutc ne '' then ran+1;

end; /* end of universe check for ran */

/* rac recode - repair and alteration costs */

if status = '1' and tenure = '1' then
do;
    rac = 0;

    if disrpam >= 2 then rac+disrpam;
    if nwbtham >= 2 then rac+nwbtham;
    if nwbedam >= 2 then rac+nwbedam;
    if nwkitam >= 2 then rac+nwkitam;
    if nwrecam >= 2 then rac+nwrecam;
    if nwotham >= 2 then rac+nwotham;

    if adbtham >= 2 then rac+adbtham;
    if adkitam >= 2 then rac+adkitam;
    if adbedam >= 2 then rac+adbedam;
    if adothing >= 2 then rac+adothing;

    if garagam >= 2 then rac+garagam;
    if porcham >= 2 then rac+porcham;
    if deckam >= 2 then rac+deckam;
    if crprtam >= 2 then rac+crprtam;

```

if othotam >= 2 then rac+othotam;

if bthwlam >= 2 then rac+bthwlam;
if bthcbam >= 2 then rac+bthcbam;
if bthflam >= 2 then rac+bthflam;
if bthctam >= 2 then rac+bthctam;
if bthtlam >= 2 then rac+bthtlam;
if bthtbam >= 2 then rac+bthtbam;
if bthskam >= 2 then rac+bthskam;
if bthltam >= 2 then rac+bthltam;
if bthelam >= 2 then rac+bthelam;
if bthptam >= 2 then rac+bthptam;

if kitwlam >= 2 then rac+kitwlam;
if kitcbam >= 2 then rac+kitcbam;
if kitflam >= 2 then rac+kitflam;
if kitctam >= 2 then rac+kitctam;
if kitapam >= 2 then rac+kitapam;
if kitskam >= 2 then rac+kitskam;
if kitltam >= 2 then rac+kitltam;
if kitelam >= 2 then rac+kitelam;
if kitptam >= 2 then rac+kitptam;

if rnbedam >= 2 then rac+rnbedam;
if rnotham >= 2 then rac+rnotham;
if roofam >= 2 then rac+roofam;

if adsidam >= 2 then rac+adsidam;
if alsidam >= 2 then rac+alsidam;
if adpipam >= 2 then rac+adpipam;
if alpipam >= 2 then rac+alpipam;
if adwiram >= 2 then rac+adwiram;
if rewiram >= 2 then rac+rewiram;
if fuseam >= 2 then rac+fuseam;
if adooram >= 2 then rac+adooram;
if rpdoram >= 2 then rac+rpdoram;
if adfxtam >= 2 then rac+adfxtam;
if rpfxtam >= 2 then rac+rpfxtam;
if adinsam >= 2 then rac+adinsam;
if rpinsam >= 2 then rac+rpinsam;
if adsubam >= 2 then rac+adsubam;
if finflam >= 2 then rac+finflam;
if adflram >= 2 then rac+adflram;
if sdflram >= 2 then rac+sdflram;
if nwpnlam >= 2 then rac+nwpnlam;
if rpanlam >= 2 then rac+rpanlam;

```

if rcenam >= 2 then rac+rcenam;
if adheam >= 2 then rac+adheam;
if hequam >= 2 then rac+hequam;
if septam >= 2 then rac+septam;
if wtrham >= 2 then rac+wtrham;
if dshwam >= 2 then rac+dshwam;
if dsplam >= 2 then rac+dsplam;
if insaam >= 2 then rac+insaam;
if insbam >= 2 then rac+insbam;
if inscam >= 2 then rac+inscam;
if driveam >= 2 then rac+driveam;
if fenceam >= 2 then rac+fenceam;
if patioam >= 2 then rac+patioam;
if poolam >= 2 then rac+poolam;
if shedam >= 2 then rac+shedam;
if outaam >= 2 then rac+outaam;
if outbam >= 2 then rac+outbam;
if outcam >= 2 then rac+outcam;
end; /* end of universe check for rac */

/* degree recode */

degree = ' ';

if (coolday <= 1999) then
do;
  if (heatday > 7000) then
    degree = '1';
  else
    if (heatday >= 5500 and heatday <= 7000) then
      degree = '2';
    else
      if (heatday >= 4000 and heatday <= 5499) then
        degree = '3';
      else
        if (heatday <= 3999) then
          degree = '4';
end;
else
  if (heatday <= 1999) then
    degree = '5';
  else
    if (heatday <= 3999) then
      degree = '6';
else
  put 'ILLEGAL DEGREE DAY VALUE -- control number = ' ctrlnm13;

```

```

/* metro recode - former geopuf */

metro = ' ';

if msasta80 = '1' then
  metro = '1';
else
  if msasta80 = '2' then
    if (uacode80 ge 1 and uacode80 le 9999) then
      metro = '2';
    else
      if urbrur80 = '1' then
        metro = '3';
      else
        metro = '4';
    else
      if urbrur80 = '1' then
        if (uacode80 ge 1 and uacode80 le 9999) then
          metro = '5';
        else
          metro = '6';
      else
        metro = '7';

/* recode for histry - set to 8 for adds and extras, otherwise set to a 1 */

srvyr = sysget('YRSURV');

histry = '1';
if convdate = srvyr then
  histry = '8';

/* recode for metro2 -- equals 1 when metro = 1, equals 2 when metro = 2-4,
*           equals 3 when metro = 5-7
***** as of May 24, 1999 this recode will not be done *****
if metro = '1' then metro2 = '1';
if '2' <= metro <= '4' then metro2 = '2';
if '5' <= metro <= '7' then metro2 = '3';           */

/* recode for metro3 -- based on msasta80 and urbrur80 only */

if msasta80 = '1' then
  metro3 = '1';
else
  if msasta80 = '2' and urbrur80 = '1' then

```

```

metro3 = '2';
else
  if msasta80 = '2' and urbrur80 = '2' then
    metro3 = '3';
  else
    if msasta80 = '3' and urbrur80 = '1' then
      metro3 = '4';
    else
      if msasta80 = '3' and urbrur80 = '2' then
        metro3 = '5';

/* finish out the top and bottom code variables: zadult, zinc, zinc2 and
then output */

if last.ctrlnm13 then
do;

  if zadult >= 11 then zadult = 11;
  if (zinc <= -10000 and zinc ne .B) then zinc = -10000;
  if zinc >= 999996 then zinc = 999996;
  if (zinc2 <= -10000 and zinc2 ne .B) then zinc2 = -10000;
  if zinc2 >= 999996 then zinc2 = 999996;

  output rcd.pufrcd;
  outcnt+1;
end;

if send then
do;
  put cnohh ' = number of persons obs without corresponding houshld obs';
  put cnopr ' = number of houshld obs without corresponding persons obs';
  put ctrlno ' = number of first control numbers processed';
  put ctllst ' = number of last control numbers processed';
  put incnt ' = number of view obs input';
  put outent ' = number of obs output';
end;

/* end of recode program */
run;

```